Recognizing Domain and Species from MEDLINE Proteomics Publications

Alexander K. Seewald[1]

Austrian Research Institute for Artificial Intelligence, Freyung 6/VI/7,
A-1010 Vienna, Austria alexsee@oefai.at

Abstract. In text mining for bioinformatics, one important bottle-neck is the availability of high-quality tagged corpora. We introduce a novel approach to learn extraction patterns from pre-classified but untagged corpora, which are easier to generate automatically. We apply our approach to two datasets derived from SWISS-PROT plus associated MEDLINE references. In both experiments a Ripper-like rule learner, JRip, is competitive to all other learners; outputs a manageable number of understandable rules; and performs comparably to a human domain expert investigating the same task. Based on our results, we note weaknesses and strengths of both human and machine learning approaches, which indicates that they have distinct areas of expertise. Our approach may be used to generate initial rulesets for information extraction, to be iteratively refined by domain experts; or as a stand-alone approach with some losses in precision.

1 Introduction

In text mining for bioinformatics, one important bottle-neck is the availability of high-quality tagged corpora. Creating a pre-tagged corpus entails high workload for domain experts, but a corpus for a specific domain can usually not be directly transferred to other domains. Disagreement between domain experts even for basic issues such as extent of protein and gene names [4] further complicate the issue.

In this paper, we propose and evaluate an alternative approach to information extraction from pre-classified, but untagged corpora; in our case created automatically from the SWISS-PROT and MEDLINE databases. We reformulate the information extraction problem as learning problem where each distinct class represents a unique slot value to be extracted from the document. Thus, we are also able to learn synonyms and utilize partial evidence for slot values. On the other hand we require that the full list of values for a given slot is known a priori. Also, the scalability of our approach towards very high number of slot values remains to be investigated.

We will proceed to show that our approach is competitive to state-of-the-art approaches such as Support Vector Machines and Decision Trees; results in a manageable number of easily understood extraction rules for each slot value and even performs competitively to a human domain expert investigating the same extraction task. The automatic approach and the domain expert each has its own area of expertise: domain experts are better at creating rules with high precision at cost of lower recall, while the automatic approach is biased towards creating rules with lower precision and higher recall. Still, the automatic approach gives better models than our domain expert in a third of cases; and generally manages the trade off between precision and recall much better.

2 Databases

SWISS-PROT [5, 1] is one of the largest proteomics databases. All its entries are created by biologists and are continually updated, extended and corrected. Thus the quality of the entries is considered to be quite high, which is not yet the case with automatically generated databases. On the other hand, the number of available examples is much smaller, but still sufficient for our experiments.

We obtained a recent snapshot of the SWISS-PROT database, consisting of 121,745 entries. We also obtained all referenced MEDLINE entries, yielding 83,051 documents. For our experiments, we focus on the OS field which encodes taxonomic information in the form of organism, or species, where the protein is present. We are interested in predicting domain and species of an organism from the associated MEDLINE documents.

All in all, there are 7,803 distinct species referenced in our SWISS-PROT snapshot, on average 1.1±0.3 per entry. It is thus not unlikely that a protein appears in more than one species; however, since then the learning problem is not well defined, we removed these entries. We also removed those entries without MEDLINE references (10.6%), and used only the first referenced MEDLINE entry[1] in each of the remaining cases, on the premise that it is the most relevant document. 104,747 entries remain after these prior selections, each consisting of a species value and an associated MEDLINE publication. These form the basis for our experiments.

From each MEDLINE publication, we chose to use title, abstract and MeSH terms. Throughout this paper, we use word occurrence vector representations.[2] We removed all characters except whitespace, lower- and uppercase letters, numbers, the dash (-) and the prime ('). Each continguous sequence of non-whitespace characters framed by whitespace[3] is considered a word. For simplicity, we combined all words from title, abstract and MeSH terms into a single word vector.

3 Species Domain

As preliminary step, we investigated the task of predicting the species domain, or kingdom – one of Archaea, Bacteria, Eukaryota or Virus – from MEDLINE

[1] On average, each SWISS-PROT entry references 1.9±2.0 MEDLINE entries.

[2] I.e. one binary attribute for each word which encodes if it appears in our text or not.

[3] ..including begin and end of the text.

Table 1. Results for predicting species domain. Acc.CV shows two-fold cross-validation on 5% of data; Acc.Test shows performance of trained model from 5% data on the remaining 95%. Approximate execution times are also given.

Classifier	Acc.CV	Exec	Acc.Test	Exec
ZeroR (baseline)	49.8%	1.4s	47.6%	0.2s
OneR	85.3%	1m	84.2%	0.1s
NaiveBayes-K	93.6%	5m	93.3%	176s
J48	96.2%	36m	96.4%	0.33s
PART	96.4%	36m	96.8%	0.4s
JRip	97.0%	108m	97.5%	0.1s
Logistic	97.4%	285m	96.4%	7s
SMO-RBF	97.5%	17m	97.6%	842s
SMO	97.7%	2m	97.8%	2.6s

documents. We chose to restrict our word occurrence vector representation to the most-frequent 1044 words, which form our attributes. As we already mentioned, this is a four-class problem.

Initially, we used 5% of our examples with two-fold cross-validation[4]. For validation, all classifiers were retrained on 5% data and evaluated on the remaining 95% for validation. A selection of common classifiers from WEKA[5] was chosen: OneR is a simple classifier which learns one rule based on a single attribute's values; NaiveBayes is another well-known classifier based on the Bayes Theorem for conditional probability; J48 is a decision tree learner based on C4.5R8, PART corresponds to c4.5rules and generates rulesets from all paths within a C4.5 decision tree; JRip is a rule learner similar in spirit to the commercial rule learner Ripper; SMO and SMO-RBF are support vector machine implementations with linear resp. Radial-Basis-Function kernels.

Results can be found in Table 1. We see that more than half of our classifiers perform similarily.[6] According to runtime, we see that the support vector implementation SMO is both much faster than JRip and slightly better; however, for the purposes of communicating our models to domain experts, JRip is much better suited. For illustration, Figure 2 shows the model which was obtained by JRip.

4 Species Top Twenty

We then decided to predict the top twenty largest species appearing in SWISS-PROT, 42.1% of all entries. This gives us 43,761 examples. We chose to restrict

[4] This is equivalent to splitting the data into two equally sized halves (retaining class distributions); using one part for training, the other part for testing; then swapping the sets, repeating train/test and averaging over the two test-set results.

[5] The source code of WEKA is available at www.cs.waikato.ac.nz/~ml/weka

[6] It deserves mention that the very simple model by OneR, *IF document contains word 'Bacterial' => Bacteria ELSE Eukaryota*, already improves significantly over the baseline.

Table 2. Results for predicting top twenty species. Acc.CV shows two-fold cross-validation on the complete dataset. Approximate execution time is also given.

Classifier	Acc.CV	Exec
`ZeroR` (baseline)	19.0%	12s
`OneR`	26.5%	52m
`NaiveBayes-K`	76.4%	10h
`JRip`	88.9%	312h
`SMO`	89.3%	29h

ourselves to the top 3,834 most frequent words. For this task, we obtained a domain expert's model for comparison. The expert utilized multi-word patterns, or phrases.[7] To ensure a fair comparison with our approach, we counted the number of rule matches and chose the rule with maximum number of matches. Ties were resolved in favor of the more common class according to training data.[8]

We chose a subset of classifiers from the previous experiment, based on considerations of runtime, accuracy and understandability. Experimental results are found in Table 2. `JRip` performs again slightly worse than `SMO`. Again, we prefer `JRip` because of its more understandable and small rule set (172 rules for twenty classes), even if its training takes an order of magnitude longer.

For a more detailed analysis, we compared each species separately, see Table 3. Maximum values are marked **bold** in this table. We can see at once that `JRip` performs better in terms of recall and F-measure[9], while the domain expert's model usually offers higher precision.

Figure 1 shows the same data in graphical form. In most cases, the domain expert improves precision at the cost of recall, i.e. the lines point[10] to the top left. Since domain experts have told us that they emphasize precision over recall, these results are not surprising. However, in seven cases `JRip` offers better precision than the human model; and in some cases the domain expert's model yields very low recall – once even less than 1%.

`JRip` also manages the trade off between precision and recall better, which can be seen by the F-measure being higher in three-fourths of cases. Average precision, recall and F-measure are also uniformly higher and the standard deviation of these values are uniformly lower than those of the domain expert, which also confirms this observation.

[7] Due to the limits of word vector representations information on word sequence is lost and thus multi-word patterns cannot be learned with our approach, which may create a significant disadvantage for our model.

[8] We expect `JRip` to use similar optimizations.

[9] The F-measure is a simple combination of recall and precision, i.e. $2 * \frac{r*p}{r+p}$.

[10] The dot at one end of the line shows the performance of our domain expert. We consider the lines to point towards this dot.

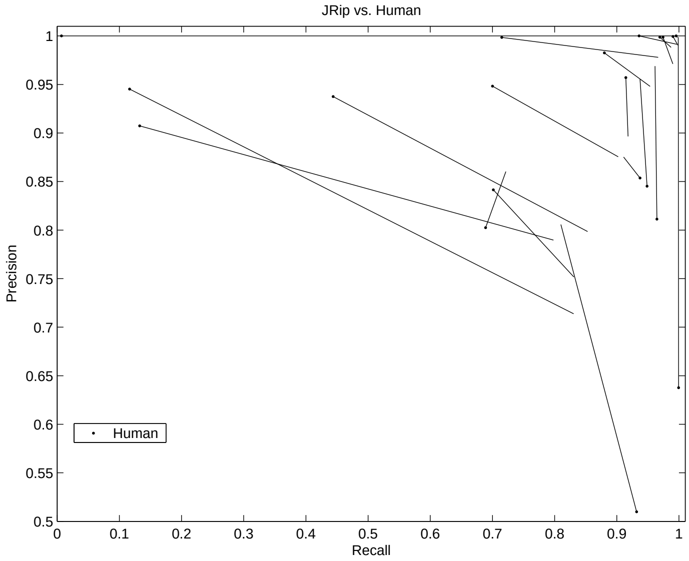

Fig. 1. Results for predicting top twenty species. X shows recall, Y shows precision. The dot (.) shows the domain expert's result, the other end of the line shows `JRip`'s result. Each line corresponds to one species.

5 Related Research

[6] introduces a system to generate extraction patterns from untagged, but preclassified corpus by exhaustively generating all extraction patterns, which are then manually selected. Our approach is related in that we extract patterns based on preclassified documents without a manually tagged corpus. No manual inspection of extraction patterns are necessary for our approach which makes it more efficient.

[7] introduces a support vector machine classifier to classify sub-cellular location. In their case, the machine learning approach still performs significantly worse than the best manually created rulesets from [3]. SVM models are also hard to visualise and understand, while our approach generates understandable rulesets of manageable size.

[2] also investigate the prediction of sub-cellular location, among other tasks. While using a pre-tagged corpus at first, they later resort to weakly labeled training instances, generated from online databases. Their approach is somewhat similar to ours, in that our training instances are also weakly labeled (i.e. we do not know where *exactly* the required information is present in the text). They also investigate using relational learning for this task, and find that it improves precision.

Table 3. Results for predicting top twenty species. p shows precision, r shows recall, F shows the F-measure, $2 * \frac{r*p}{r+p}$. On the left, we see results by `JRip`, on the right those from the domain expert's model. Best values are shown in **bold**. Average and standard deviation for all columns is also given.

Species Name	JRip			Human		
	p	r	F	p	r	F
Homo sapiens (Human)	**80.57**	81.00	**80.79**	50.99	**93.18**	65.91
Xenopus laevis (African clawed frog)	75.17	**83.10**	**78.94**	**84.15**	70.14	76.51
Escherichia coli	**96.90**	96.13	**96.51**	81.13	**96.45**	88.13
Caenorhabditis elegans	**87.53**	91.09	89.27	85.37	**93.73**	**89.36**
Haemophilus influenzae	99.04	**99.83**	99.43	**99.94**	99.03	**99.49**
Arabidopsis thaliana (Mouse-ear cress)	97.80	**96.66**	**97.23**	**99.85**	71.49	83.32
Bos taurus (Bovine)	71.38	**83.05**	**76.77**	**94.52**	11.64	20.72
Bacillus subtilis	98.82	**98.74**	**98.78**	**99.87**	96.93	98.38
Archaeoglobus fulgidus	**100.00**	**100.00**	**100.00**	**100.00**	0.69	1.36
Mus musculus (Mouse)	78.97	**79.81**	**79.39**	**90.73**	13.26	23.15
Mycobacterium tuberculosis	99.85	**99.77**	**99.81**	**100.00**	99.55	99.77
Salmonella typhimurium	89.65	**91.81**	90.71	**95.71**	91.45	**93.53**
Synechocystis sp. (strain PCC 6803)	99.13	**99.67**	**99.40**	**100.00**	93.57	96.68
Pseudomonas aeruginosa	97.12	**99.02**	98.06	**99.87**	97.43	**98.64**
Saccharomyces cerevisiae (Bakers yeast)	**95.55**	93.73	**94.63**	84.52	**94.85**	89.39
Methanococcus jannaschii	**99.87**	99.87	**99.87**	63.76	**99.93**	77.85
Gallus gallus (Chicken)	79.85	**85.31**	**82.49**	**93.75**	44.39	60.25
Rattus norvegicus (Rat)	**86.04**	**72.16**	**78.49**	80.24	68.90	74.14
Schizosaccharomyces pombe (Fission yeast)	94.81	**95.35**	**95.08**	**98.25**	87.99	92.84
Drosophila melanogaster (Fruit fly)	87.55	**90.23**	**88.87**	**94.83**	70.00	80.55
Max	7	**15**	**16**	**14**	5	4
Avg	**90.78**	**91.82**	**91.23**	89.87	74.73	75.50
± stdDev	**9.31**	**8.35**	**8.62**	13.22	32.03	28.56

6 Conclusion

We have reformulated a problem of information extraction within BioInformatics as learning problem. More specifically, we have aimed to extract organism names (domain and species) from MEDLINE documents related to proteomics.

The reported results show that our approach is competitive to a human domain expert, both having distinct areas of expertise: Humans are better at creating rules with high precision at cost of lower recall, while our approach is well suited to create rules with lower precision and higher recall. This is in accordance with our domain expert's preference for precision. Still, in a third of cases our approach yields models with better precision than the domain expert's models. The focus on precision over recall may impair the domain expert's ability for a reasonable trade off between precision and recall[11] while our approach manages the trade off fairly well.

[11] Recall as low as 0.7% was observed, for small or even nonexisting improvements in precision

```
(archaeon) => domain=A (163.0/0.0)
(Archaeal) and (!Bacterial) => domain=A (92.0/0.0)
(Halobacterium) => domain=A (22.0/3.0)
(archaebacterium) and (Bacterial) => domain=A (7.0/0.0)
(Methanobacterium) => domain=A (6.0/2.0)
(Archaea) and (!Proteins) => domain=A (2.0/0.0)
(Viral) => domain=V (351.0/18.0)
(Bacterial) and (!Animal) => domain=B (1665.0/14.0)
(Bacterial) and (!RNA) and (!cerevisiae) => domain=B (211.0/10.0)
(!Animal) and (Escherichia) and (!Proteins) and (!Fungal) and (!cDNA) => domain=B (26.0/2.0)
(!Animal) and (bacteria) and (!cDNA) => domain=B (19.0/3.0)
(strain) and (!Fungal) and (!Proteins) and (!2) => domain=B (17.0/1.0)
(!Animal) and (cyanobacterium) => domain=B (9.0/1.0)
(Bacteria) and (!Animal) => domain=B (6.0/1.0)
(Frames) and (operon) => domain=B (4.0/1.0)
(Salmonella) => domain=B (2.0/0.0)
(Streptomyces) and (!at) => domain=B (5.0/0.0)
(Anabaena) => domain=B (3.0/0.0)
(bacterium) => domain=B (5.0/1.0)
(Bacillus) and (!Animal) => domain=B (5.0/1.0)
(pneumoniae) => domain=B (2.0/0.0)
 => domain=E (2534.0/20.0)
```

Fig. 2. `JRip`'s model for species domain. (word) encodes word occurrence and (!word) word non-occurrence. E.g. second rule reads like this: If the (title, abstract, MeSH terms) of a MEDLINE entry contains *Archaeal* and not *Bacterial*, predict domain archaea (domain=A). This rule is corrent for 92 examples and incorrect for none (92.0/0.0). The last line is the default rule, which is chosen if no other rule matches.

Our approach returns an easily understandable ruleset of manageable size[12] and could either be used for generating initial rulesets for iterative refinement by domain experts; or as a stand-alone approach with some losses in precision.

We intend to investigate the scalability of our approach to thousands of slot values in the future. We will also look into relational learning approaches and support vector machines with string kernels; and other interesting problems within the field BioInformatics.

Acknowledgements

This research was supported by the European Union under project no. QLRI-2002-02770 (BioMinT). The Austrian Research Institute is supported by the Austrian Federal Ministry of Education, Science and Culture. We want to thank our domain expert Violaine Pillet from the Swiss Institute of Bioinformatics for sharing with us her extraction rules for SWISSPROT species.

References

1. Boeckmann B., Bairoch A., Apweiler R., Blatter M.-C., Estreicher A., Gasteiger E., Martin M.J., Michoud K., O'Donovan C., Phan I., Pilbout S., Schneider M. (2003) The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003, Nucleic Acids Research, 31(1):365-370.

[12] 22 rules for species-domain, see Figure 2; 172 rules for species-top20.

2. Craven M., Kumlien J. (1999) Constructing Biological Knowledge Bases by Extracting Information from Text Sources, Proceedings of the 7th International Conference on Intelligent Systems for Molecular Biology (ISMB-99).
3. Eisenhaber F., Bork P. (1999) Evaluation of human-readable annotation in biomolecular sequence databases with biological rule libraries, BioInformatics (15) 7-8, pp.528-535.
4. Franzen K., Eriksson G., Olsson F., Asker L., Liden P., Coester J. (2002) Protein names and how to find them, International Journal of Medical Informatics, Vol. 67/1-3, Special Issue on NLP in Biomedical Applications, pp.49-61.
5. O'Donovan,C., Martin,M.J., Gattiker,A., Gasteiger,E., Bairoch,A. and Apweiler,R. (2002) High-quality protein knowledge resource: SWISS-PROT and TrEMBL. Briefings in Bioinformatics, *3*, 275–284.
6. Riloff,E. (1996) Automatically Generating Extraction Patterns from Untagged Text, in Proceedings of the 13th National Conference on Artificial Intelligence, AAAI Press/MIT Press, Cambridge/Menlo Park, pp.1044-1049.
7. Stapley B.J., Kelley L.A., Sternberg M.J.E. (2002) Predicting the Sub-Cellular Location of Proteins from Text Using Support Vector Machines, Pacific Symposium on Biocomputing (7), pp.374-385.